This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Withdrawn) An authentication system, comprising:
a filter to monitor sessions between a client and a server for proper authentication;
a plug-in coupled to the client and the server, said plug-in to generate public and private key pairs, and to receive and store certificates; and
an extension coupled to said filter, said extension to generate script commands to cause the client and the server to perform required operations indicated by said filter.

2. (Withdrawn) The system of claim 1, wherein the certificates are used to certify the client to the server.

3. (Withdrawn) The system of claim 1, wherein the certificates are used to certify the server to the client.

4. (Withdrawn) The system of claim 1, wherein the certificates are used to certify the client and the server to each other.

5. (Withdrawn) The system of claim 1, wherein the script commands are implemented in a hypertext markup language (HTML) program.

6. (Withdrawn) A secure client/server system, comprising:
a client to request data or service;
a server to provide the requested data or service; and
an authentication system including:

a filter to monitor sessions between the client and the server for proper authentication,

a plug-in coupled to the client and the server, said plug-in to generate public and private key pairs, and to receive and store certificates, and

an extension coupled to said filter, said extension to generate script commands to cause the client and the server to perform required steps indicated by said filter.

7. (Withdrawn) The system of claim 6, wherein the certificates are used to certify the client to the server.

8. (Currently Amended) A method for providing a single sign-on authentication and privacy, comprising:

submitting a request to access a node;

directing to submit a certificate;

verifying the submitted certificate with a trusted certificate, wherein the verifying step is performed by a security extension in a server and operates to verify a certificate sent from a client to the server;

performing a challenge, wherein the challenge is generated by ~~a~~ the security extension ~~in a server~~ and is sent to the client;

generating a response to the challenge; and

saving the response as a named cookie.

9. (Original) The method of claim 8, wherein said response is used as a security token.

10. (Original) The method of claim 9, wherein said security token is used to propagate an initial authentication.

11. (Original) The method of claim 8, further comprising:
creating a connection session if the certificate is valid.

12. (Previously Presented) The method of claim 8, wherein said verifying the submitted certificate includes checking a signature on the submitted certificate with the trusted certificate.

13. (Previously Presented) The method of claim 8, further comprising:
generating a key;
encrypting the key with a client's public key;
sending an encrypted key to a client; and
using the key to encrypt communication.

14. (Withdrawn) A method for providing client privacy, comprising:
determining the identity of a client;
generating a key, wherein the key is a symmetric key generated by a security filter;
encrypting the key with a client's public key;
sending an encrypted key to a client; and
using the key to encrypt communication.

15. (Withdrawn) The method of claim 14, wherein said sending the encrypted key includes sending the key using a hypertext transfer protocol (HTTP) header.

16. (Currently Amended) A method for providing a single sign-on authentication and privacy, comprising:
submitting a request to access a node;
directing to submit a certificate;

verifying the submitted certificate with a trusted certificate, wherein the verifying step is performed by a security extension in a server and operates to verify a certificate sent from a client to the server;

performing a challenge, wherein the challenge is generated by ~~a~~ the security extension ~~in a server~~ and is sent to the client;

generating a response to the challenge;

saving the response as a named cookie with an authentication token; and

using standard Secure Socket Layer (SSL) library to provide communication privacy.

17. (Previously Presented) The method of claim 16, wherein said verifying includes creating and registering a new authentication session.

18. (Original) The method of claim 17, wherein said verifying includes validating the new authentication session with the authentication token.

19. (Original) The method of claim 16, wherein said verifying includes indicating a failure status to a client if said verifying fails.

20. (Original) The method of claim 16, wherein said performing said challenge includes generating a node challenge random number.

21. (Original) A method of claim 16, wherein said directing includes receiving an address of the node; and

checking to determine if the address is protected.

22. (Original) The method of claim 16, further comprising:

determining if the authentication token is already present.

23. (Previously Presented) The method of claim 22, further comprising:

determining if a client is on an access control list if the authentication token is present and valid.

24. (Currently Amended) An apparatus comprising a computer-readable storage medium having executable instructions that enable the computer to:

submit a request to access a node;

direct to submit a certificate;

verify the submitted certificate with a trusted certificate, wherein such verifying is performed by a security extension in a server and operates to verify a certificate sent from a client to the server;

perform a challenge, wherein the challenge is generated by a the security extension ~~in a server~~ and is sent to the client;

generate a response to the challenge; and

save the response as a named cookie.

25. (Original) The apparatus of claim 24, wherein said response is used as a security token.

26. (Currently Amended) An apparatus comprising a computer-readable storage medium having executable instructions that enable the computer to:

submit a request to access a node;

direct to submit a certificate;

verify the submitted certificate with a trusted certificate, wherein such verifying is performed by a security extension in a server and operates to verify a certificate sent from a client to the server;

perform a challenge, wherein the challenge is generated by ~~a~~ the security extension ~~in a server~~ and is sent to the client;

generate a response to the challenge;

save the response as a named cookie with an authentication token; and

use standard Secure Socket Layer (SSL) library to provide communication privacy.

27. (Original) The apparatus of claim 26, wherein said verify the submitted certificate includes instructions to create and register new authentication session.